IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 1, November 2023

Malicious Application Detection in Windows Using Machine Learning

Sanskar Patil[1], Mrs. Swati Jakkan[2], Sumit Pawar[3], Prerana Gholve[4], Aditi Kunkekar[5]
Faculty, Department of Computer Engineering [2]
Student, Department of Computer Engineering [1,3,4,5]
RMD Sinhgad School of Engineering, SPPU, Pune, India

Abstract: *As the proliferation of digital technology continues, the threat landscape for Windows operating systems has become increasingly complex. Malicious applications, including viruses, ransomware, and spyware, pose a significant risk to both individuals and organizations. To combat this growing threat, there is a pressing need for effective and efficient methods for detecting and mitigating malicious applications. This research paper presents an innovative approach to Malicious Application Detection in Windows using Support Vector Machine (SVM) algorithms. SVM is a powerful machine learning technique that has been successfully applied in various classification tasks, including malware detection. The primary objective of this study is to develop a robust and reliable system that can differentiate between benign and malicious applications in a Windows environment. We start by collecting a comprehensive dataset of Windows applications, comprising both legitimate and malicious software samples. Feature extraction techniques are employed to convert the application data into a suitable format for SVM analysis. These features may include file attributes, system call sequences, and behaviour analysis metrics.*

Keywords: Malicious Application Detection, Machine Learning Based Detecting, Windows Malware Detection, Windows Security.

I. INTRODUCTION

Malicious application detection is a critical aspect of computer security, especially in the Windows operating system environment, which is a prime target for various forms of malware and malicious software. One effective method for detecting and classifying malicious applications is through the use of machine learning algorithms, with Support Vector Machines (SVM) beinga popular choice. In this introduction, we will discuss the importance of detecting malicious applications in Windows and how SVM can be leveraged for this purpose.

II. OBJECTIVE

We employ the Windows Dataset to find malicious software. We input a Windows data collection, which is then subjected to preprocessing. After both phases are complete, we employ the SVM technique to classify and detect malicious applications during the segmentation step.

III. LITERATURE SURVEY

Sr.No	Publication Detail	Methodology Algorithm	Dataset	Accuracy	Research Gap Identification
1	Detecting Malicious Attacks Exploiting Hardware Vulnerabilities Using Performance Counters	Machine Learning Classifiers	Android Malware Dataset for Machine Learning	Promising accuracy (exact value not provided)	Hardware-level vulnerability detection
2	Detection of Malware Using SVM	Linear SVM	Malware Classification Dataset	72-86 %	Not specified
3	Practical Automated	Not specified	Not specified	80%	Automated Detection

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 1, November 2023

	Detection of Malicious npm Packages				of Malicious npm Packages
4	Significance of Machine Learning for Detection of Malicious Websites on an Unbalanced Dataset	ARF	Unbalanced dataset	92% to 95%	The significance of machine learning in addressing unbalanced datasets
5	Efficient Intrusion Detection of Malicious Node Using Bayesian Hybrid Detection in MANET	Bayesian Hybrid Detection	Not specified	95%	Combining anomaly-based and signaturebased intrusion detection in MANETs
6	Detection of Malicious Software by Analyzing Distinct Artifacts Using Machine Learning and Deep Learning Algorithms	Machine Learning and Deep Learning Algorithms	Not specified	97%	Leveraging machine learning and deep learning for malware detection
7	Detecting Malicious Android Applications Based on API Calls and Permissions Using Machine Learning Algorithms	Na	Datasetfeaturescategories	96%	Comparing and analyzing different Android malware detection systems
8	Resilient Consensus of Multiagent Systems Under Malicious Attacks: Appointed-Time Observer-Based Approach	Appointed-Time Observer-Based Approach	Not specified	Not specified	Achieving resilient consensus in multiagent systems
9	Detecting Malicious Attacks Exploiting Hardware Vulnerabilities Using Performance Counters	Machine Learning Classifiers	Datasetfeaturescategories	96%	Hardware-level vulnerability detection
10	Security to Wireless Sensor Networks Against Malicious Attacks Using Hamming Residue Method	Hamming Residue Method (HRM)	Malware Classification Dataset	50%	Not specified

IV. ALGORITHMIC SURVEY

Sr no	Algorithm Name	Strategy used	Accuracy	Time complexity	Space complexity
1	Signature based detection	Pattern matching	90%	O(n * m)	O(m)

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 1, November 2023

2	Anomaly Detection	Baseline Deviation	75 % to 90 %	O(n *log(n))	O(n)
3	Behavioural analysis	Learning From Behaviour	92%	O(n *log(n))	O(n)
4	Feature Extraction	Feature Engineering	75% to 89%	O(m * log(m))	O(m)
5	Deep learning	Neural network	95%	O(m * log(m)	O(m)
6	Natural language processing (NLP)	Text analysis	85% to 92%	O(n*log(n))	O(n)
7	Reinforcement learning	Dynamic dessison optimization	75% to 89%	O(k * log (k))	O(k)
8	Deep learning for sequence data	LSTM and sequence analysis	80% to 95%	O(p * log(p)	O(p)

V. CONCLUSION

A project focused on malicious application detection in Windows using the Support Vector Machine (SVM) algorithm is a critical initiative in the realm of cyber security. Such a project seeks to address the evolving threats posed by malicious software to Windows-based systems, aiming to enhance security, protect user data, and ensure the integrity of Windows environments. SVM is a popular choice for this task due to its ability to create an optimal hyperplane that maximizes the margin between different classes of data, making it effective for binary classification problems like benign vs. malicious application detection. However, it's worth noting that this is just one approach, and there are other machine learning and deep learning methods used for similar purposes in the field of cybersecurity.

REFERENCES

[1]. Mahmoud Alfadel, Diego Elias Costa, and Emad Shihab. 2021. Empirical Analysis of Security Vulnerabilities in Python Packages. In 2021 IEEE International Conference on Software Analysis, Evolutionand Reengineering(SANER).446–457. https://doi.org/10.1109/SANER50967.2021.00048

[2]. Vitalii Avdiienko, Konstantin Kuznetsov, Alessandra Gorla, Andreas Zeller, Steven Arzt, Siegfried Rasthofer, and Eric Bodden. 2015. Mining Apps for Abnormal Usage of Sensitive Data. In 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, Vol. 1. 426–436. https://doi.org/10.1109/ICSE.2015.61

[3]. Aadesh Bagmar, Josiah Wedgwood, Dave Levin, and Jim Purtilo. 2021. I Know What You Imported Last Summer: A study of security threats in the Python ecosystem. CoRR abs/2102.06301 (2021). arXiv:2102.06301 https://arxiv.org/abs/ 2102.06301

[4]. Adam Baldwin. 2019. Plot to steal cryptocurrency foiled by the npm security team. https://blog.npmjs.org/post/185397814280/plotto-steal-cryptocurrencyfoiled-by-the-npm.

[5]. Alex Birsan. 2021. Dependency Confusion: How I Hacked Into Apple, Microsoft and Dozens of Other Companies. https://medium.com/@alex.birsan/ dependency-confusion-4a5d60fec610

[6]. Mengdi Huai, Di Wang, Chenglin Miao, and Aidong Zhang. 2020. TowardsInterpretationofPairwiseLearning. In Thirty-fourth AAAI Conference on Artificial Intelligence.

[7]. Sandy Huang, Nicolas Papernot, Ian Goodfellow, Yan Duan, and PieterAbbeel. 2017. Adversarial attacks on neural network policies. arXiv preprint arXiv:1702.02284 (2017).

[8]. L´eonardHussenot, Matthieu Geist, and Olivier Pietquin. 2019. Targeted Attacks on Deep Reinforcement Learning Agents through Adversarial Observations. arXiv preprint arXiv:1905.12282 (2019).

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
Impact Factor: 7.301
Volume 3, Issue 1, November 2023

[9]. Rahul Iyer, Yuezhang Li, Huao Li, Michael Lewis, Ramitha Sundar, and Katia Sycara. 2018. Transparency and explanation in deep reinforcementlearningneuralnetworks. InProc. OftheAAAI/ACM Conference on AI, Ethics, and Society.

[10]. Michael Kearns and Satinder Singh. 2002. Near-Optimal Reinforcement Learning in Polynomial Time. Mach. Learn. (2002).